不同文件格式的压缩性能分析

杨含[1]　　秦广军[*]　　胡永庆[1]
[1]（北京联合大学智慧城市学院 北京 100101）
[*]（通讯作者 Email:zhtguangjun@buu.edu.cn）

摘要：在数据存储与传输中，文件压缩是减少数据量的常用技术，可减少数据存储空间和传输时间及带宽。然而，不同类型文件格式的压缩性能存在显著差异，收益也不同。本文收集 22 种文件格式，约 178GB 数据，采用 Zlib 算法进行压缩实验来比较性能，以研究不同文件类型的压缩收益。实验结果发现，某些文件类型的压缩效果较差，压缩后文件大小几乎不变，压缩时间长，收益较低；另一部分文件类型经过压缩后文件大小明显减小，压缩时间也较短，可以有效降低数据量。基于上述实验结果，本文后续将在数据存储和传输中针对文件类型有选择性的通过压缩来减少数据量，以获得最大压缩收益。

关键词：文件压缩 传输时间 文件格式 压缩性能 Zlib 算法

Compression performance analysis of different file formats

Yang Han[1]　Qin GuangJun[*]　Hu YongQing[1]
[1](Smart City College, Beijing Union University, Beijing 100101, China)
[*](Corresponding author(s). Email(s):zhtguangjun@buu.edu.cn)

Abstract: In data storage and transmission, file compression is a common technique for reducing the volume of data, reducing data storage space and transmission time and bandwidth. However, there are significant differences in the compression performance of different types of file formats, and the benefits vary. In this paper, 22 file formats with approximately 178GB of data were collected and the Zlib algorithm was used for compression experiments to compare performance in order to investigate the compression gains of different file types. The experimental results show that some file types are poorly compressed, with almost constant file size and long compression time, resulting in lower gains; some other file types are significantly reduced in file size and compression time after compression, which can effectively reduce the data volume. Based on the above experimental results, this paper will then selectively reduce the data volume by compression in data storage and transmission for the file types in order to obtain the maximum compression yield.

Keywords: File compression　Transfer time　File format　Compression performance　Zlib algorithm

1　引言

随着计算机技术在各领域的广泛应用，产生了大量需要存储、计算和传输的数据，数据规模逐年呈爆炸式增长，表明已经踏入海量数据时代[1]。这些海量数据都需要快速迁移到计算和存储设备，导致数据传输与业务需求之间的矛盾日益尖锐[6]，从带宽需求到传输完整性，均面临重大挑战[5]。

提高海量数据传输性能的有效办法之一是降低数据规模，即通过数据压缩传

输来减少网络负载和传输延迟。通过压缩数据大小，将数据转换成更紧凑的格式，减少有效传输数据量，同时也可减少传输所需的时间和存储空间[4]，从而降低传输延迟和成本，提升数据传输速度[2]。但是，不同文件格式的可压缩性存在较大差异，某些文件格式具有高度可压缩的结构，例如文本文件中的重复字符可以被充分利用以实现更好的压缩效果。相比之下，图像文件的相邻像素通常相似且存在冗余，但对已经压缩过的图像文件再次压缩的效果有限。音频文件通常采用有损压缩算法，因此对其进行进一步的压缩效果较差。因此，为了研究数据传输中不同文件格式的压缩收益，提高数据存储和传输性能。本文在 22 种不同文件格式上进行了压缩实验，以研究不同文件格式的压缩收益，为海量数据传输提供参考依据。

本文的贡献如下：

1. 收集了 22 种文件格式，包括 MP4、MP3、BMP、HDF5 等，共计 178GB；
2. 采用 zlib 压缩算法研究了上述数据集；
3. 实验中发现：一些文件类型，如音频、视频和图像等，其压缩效果相对较差，压缩时间较长，从而导致压缩带来的收益相对较低。这是由于这些文件类型通常具有高维度的数据结构和复杂的信息内容，使得在压缩过程中难以实现较高的压缩率。然而，并非所有文件类型都面临相同的问题。对于一些其他类型的文件，如文本、文档等，经过压缩后文件大小明显减小，压缩时间也较短，可以有效降低数据量。

本文后续的章节结构如下：

1. 第 2 节描述了有关文件压缩的相关工作；
2. 第 3 节描述了实验数据集文件来源、压缩性能指标和实验方法；
3. 第 4 节描述实验环境和工具；
4. 第 5 节对实验结果进行图表分析；
5. 第 6 节进行总结和展望。

2 相关工作

文件压缩是重要的数据处理技术，通过减小文件的存储大小，降低磁盘空间占用，提高存储效率和传输速度。压缩算法可分为无损压缩和有损压缩，前者完全恢复原始文件，后者在一定程度上损失信息以获得更高的压缩率。多媒体信息常采用有损压缩，文本文件需要完整性的文件则采用无损压缩算法。Gzip[17]是广泛使用的压缩程序，可用于压缩大的、较少使用的文件以节省磁盘空间，其压缩比率在 3 到 10 倍左右，可显著减少服务器的网络带宽消耗。bzip2[18]采用 Burrows-Wheeler 块排序文本压缩算法和 Huffman 编码方式，压缩率通常优于基于 LZ77/LZ78 的压缩软件，可将文件压缩至 10%至 15%以内。Lzma[7]是经过改良和优化的 Deflate 和 LZ77 算法，采用类似于 LZ77 的字典编码机制，在一般情况下具有比 bzip2 更高的压缩率。Zlib 库[19]提供了高压缩比和无损压缩功能，使用基于滑动窗口机制的 DEFLATE 算法，以字节为单位处理数据，通过替换字符串来实现压缩，在多个领域如中文检索、数据通讯和数据采集等中广泛应用[3]。

在应用中，李明等人采用基于信号稀疏表示的无损压缩传输算法，提升了单位时间内的上传信息量[40]。王巨龙等人利用 Steim2 压缩算法和 FTP 通信协议实现了实时数据压缩和传输，显著提高了数据传输效率[41]。杨敬锋等人提出了基于改进 Huffman 编码技术的数据压缩方法，实现了数据的压缩、传输、解析和解压[43]。彭冲等人提出了基于节点相似性分簇的压缩方案（SSCDCT），通过

聚集相似节点和压缩算法减少了数据传输量和能耗，延长了网络寿命[44]。马兴明等人提出了基于状态估计的海量多元异构智能电网数据压缩存储方法，解决了压缩误差大和运行时间长的问题[45]。王鹤等人提出了一种基于分布式压缩感知和边缘计算的电能质量数据压缩存储方法，解决了电能质量数据和谐波污染划分困难的问题，实现了高精度压缩和节省存储空间[46]。

可见，在海量数据存储和传输领域，压缩技术具有显著的效益。然而，对于不同格式的海量文件，其压缩效益尚未经过系统的研究和评估。因此，本实验的开展具有重要意义。本文的研究目标是探究不同格式的海量文件在解压缩方面的效益，以实现后续存储和传输过程的优化。通过本文的研究成果，将为数据管理和传输技术领域提供有价值的见解和指导。

3 方法

3.1 文件数据集选择和准备

本实验使用了多个数据集来形成不同格式的数据集，数据来源如下：

1. 视频格式 MP4，来自 KAGGLE 网站的 Kinetics dataset 数据集。
2. 视频格式 AVI、MKV、WEBM，通过“格式工厂[1]”在 Kinetics dataset 基础上扩增。
3. 音频格式 MP3，来自 UCI 网站的数据集 FMA:A Dataset For Music Analysis Data Set。
4. 音频格式 FLAC、WAV、WMA，通过格式工厂转换自数据集 FMA: A Dataset For Music Analysis DataSet。
5. 图像格式 BMP，来自 KAGGLE 网站的 Alphabet+Numbers。
6. 图像格式 GIF，来自 KAGGLE 网站的 Synthea Dataset Jsons – HER。
7. 图像格式 JPG，来自 CVPR2015 论文 A Large-Scale CarDataset for Fine-Grained Categorization and Verification 的数据集。
8. 图像格式 PNG，来自 KAGGLE 网站的数据集 RSNA BreastCancer Detection - 512×512 pngs
9. 图像格式 TIF，通过格式工厂转换自 RSNA BreastCancer Detection - 512×512 pngs。
10. 文档格式 DOCX、XLS、XML 数据集为自采集。
11. 文档格式 PDF，通过格式工厂转换自自采的 DOCX 数据集。
12. 文档格式 TXT，来自 KAGGLE 网站数据集 Text Classification on Emails。
13. 文档格式 JSON，来自 KAGGLE 网站数据集 Various Pokemon Image Dataset。

实验数据集文件如表 1 所示：

表 1 实验数据集文件

数据集名称	文件格式	文件类型
转自Kinetics dataset	视频	AVI
转自Kinetics dataset	视频	MKV
Kinetics dataset[10]	视频	MP4
转自Kinetics dataset	视频	WEBM

1 格式工厂-免费多功能的多媒体文件转换工具(formatfactory.org)

转自FMA	音频	FLAC
FMA: A Dataset For Music Analysis Data Set[11]	音频	MP3
转自FMA	音频	WAV
转自FMA	音频	WMA
Metal Surface Defects Dataset[15]	图像	BMP
Synthea Dataset Jsons - EHR[12]	图像	GIF
A Large-Scale Car Dataset for Fine-Grained Categorization and Verification. (CVPR)[14]	图像	JPG
RSNA Breast Cancer Detection - 512x512 pngs[9]	图像	PNG
转自A Large-Scale Car Dataset for Fine	图像	TIF
转自DOCX	文档	PDF
自采	文档	DOCX
自采	文档	XLS
自采	文本	XML
Text Classification on Emails[13]	文本	TXT
Various Pokemon Image Dataset[8]		JSON
自采	二进制	BIN

在实验中，使用自采数据集和公开数据集共 91GB，对不同文件格式进行了压缩效果的评估。这些公开数据集涵盖了 20 种文件类型，包括文本文件、图像文件、音频文件和视频文件等。

3.2 压缩性能评估指标

令压缩前的文件大小为α，压缩后的文件大小为β，压缩率为 K，压缩时间为 T，解压时间为 T’, 则压缩率指标可表示为:

$$K=(1-\beta/\alpha)*100\%$$

• 压缩率:表示压缩减少的数据比例，计算每个文件类型的平均压缩率，并比较不同文件类型之间的压缩率差异。K 的值越大，表示压缩效果越好。

• 压缩时间:表示原始文件压缩为压缩文件所花费的时间，以毫秒为单位。计算每个文件类型的平均压缩时间，并比较不同文件类型之间的压缩时间差异。T 的值越小，说明压缩越快。

• 解压时间:表示压缩文件被解压缩为原始文件所花费的时间，以毫秒为单位。计算每个文件类型的平均解压缩时间，并比较不同文件类型之间的解压缩时间差异。T’的值越小，说明解压越快。

3.3 压缩方法

在选择适合的文件压缩算法时，需要根据需求和场景进行比较。Zlib 库以其高压缩比和无损压缩的优势，通过 DEFLATE 算法实现了最大程度的文件大小减小。特别适用于网络传输，能够有效降低带宽消耗。Zlib 拥有广泛的支持和跨平台性，方便调用其函数和接口进行文件压缩和解压缩操作，这使得 Zlib 成为进行实验的理想选择，以获得更准确和全面的实验结果。因此，本实验选用 Zlib 函数库，对参数进行如下定义：压缩前的文件为 X，压缩后的文件为 Y。

压缩过程可表示为：Y=Zlib(X)

解压过程可表示为：X=Zlib(Y)

实验步骤如下：

1. 导入测试数据集；

2. 调用 Zlib 库遍历指定目录下的文件列表，对每个文件样本进行压缩过程 **Y=Zlib(X)**和解压缩过程 **X=Zlib(Y)**，并记录压缩比率 K、压缩时间 T、解压时间 T'、文件压缩前大小α、压缩后大小β等信息等性能指标；

3. 删除过程中生成的压缩文件和解压缩文件；

4. 将将结果记录写入 CSV 文件；

5. 在 CSV 文件中计算每项指标的平均值以提高结果的可靠性。

4 实验

在实验中，首先对 20 种文件数据集进行了压缩和解压缩，并对结果进行了详尽的分析。为了确保实验结果的准确性，在压缩和解压后立即删除生成的文件，以避免占用额外的存储空间。本实验使用 3.2 节的压缩性能指标作为评价标准。

4.1 实验环境和工具

本实验在 64 位的 Ubuntu 22.04.2LTS 计算机上进行，实验环境配置了 32.0GIB 的内存和 1T 的磁盘容量，g++-11 编译器。

4.2 实验分析

（1）全量数据集

对全部 20 种不同类型的文件数据集使用了 Zlib 库进行压缩和解压缩操作，并记录了每个文件类型的压缩时间解压缩时间、解压前文件大小、解压后文件大小、压缩率。实验所使用的数据集约 91GB 数据，视频文件（AVI、MKV、MP4、WEBM）约共 54.4GB、音频文件（FLAC、MP3、WAV、WMA）约共 12.58GB、图像文件（BMP、GIF、JPG、PNG、TIF）约共 9.65GB、文档文件（PDF、DOCX、XLS）约共 0.96GB、文本文件（XML、TXT）约共 0.18GB、JSON 文件约 0.85GB、BIN 文件约 0.04GB。

实验结果如表 2 所示：

表 2 实验数据结果

参数 / 类型	数量（个）	压缩前大小 (MB)	压缩后大小 (MB)	压缩时间 (MS)	解压时间 (MS)	压缩率 (%)
AVI	12748	0.522941	0.508537	17.76	3.28	3.49%
MKV	10350	2.105637	2.102279	54.733	6.656	0.71%
MP4	10214	1.469854	1.463328	38.026	3.997	0.00
WEBM	6653	1.200335	1.20309	33.938	2.162	0.12%
FLAC	8733	0.581497	0.574143	13.019	0.963	1.93%
MP3	8732	0.13802	0.134574	4.877	0.972	2.98%
WAV	8733	0.774194	0.660883	30.932	5.777	14.02%
WMA	8733	0.242912	0.109045	5.96	1.326	54.61%
BMP	15557	0.750051	0.132008	22.156	3.6	82.40%
GIF	7261	0.08965	0.086222	3.781	0.678	4.16%
JPG	10547	0.089225	0.088777	2.693	0.392	0.66%
PNG	54707	0.065476	0.065365	1.996	0.287	0.28%
TIF	8895	0.140895	0.135055	4.835	0.981	5.00%

PDF	2485	0.02381	0.021898	1.193	0.225	8.08%
DOCX	2485	0.051761	0.043766	2.193	0.389	14.75%
XLS	15121	0.052615	0.012066	2.582	0.324	77.05%
XML	3912	0.043191	0.003706	1.129	0.19	89.51%
TXT	7760	0.002006	0.001005	0.302	0.67	42.28%
JSON	10628	0.080233	0.019754	1.926	0.399	68.64%
BIN	1500	0.029862	0.003754	1.861	0.198	86.93%

20 种格式的文件压缩率、压缩时间、解压时间对比如图 1 所示：

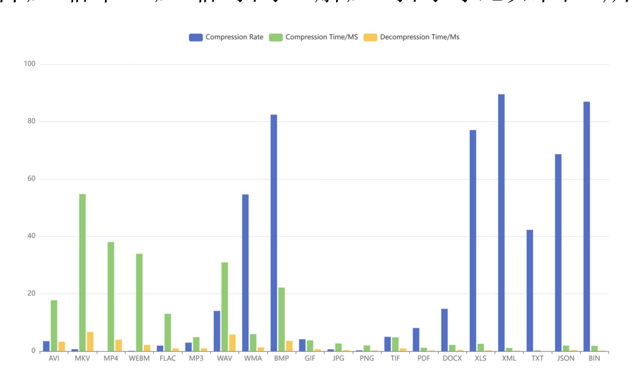


图 1 20 种文件格式压缩率、压缩时间、解压时间比较

实验结果表明不同文件类型的压缩率表现出明显差异，视频格式（AVI、MKV、MP4、WEBM）、音频格式（FLAC、MP3）、图像格式（GIF、JPG、PNG、TIF）在经过压缩后，其文件大小几乎没有明显变化，收益较低；相反，其他文件类型在经过压缩后显著减小，压缩时间也较短，可以有效降低数据量。具体实验分析如下：

1. 可以观察到不同文件类型的压缩时间也表现出明显差异。例如，视频文件如：MKV 文件（54.733 毫秒）和 MP4 文件（38.026 毫秒）的压缩时间较长；而文本文件如：TXT 文件（0.302 毫秒）和 XML 文件（1.129 秒）显示出较短的压缩时间，这是因为视频文件通常具有较大的文件大小，而文本文件通常较小。由于压缩算法需要处理更多的数据块和复杂的数据结构，因此处理较大文件所需的时间相应较长。相比之下，较小的文本文件由于其相对简单的结构，可以更快地进行压缩处理。
2. 可以观察到不同文件类型的解压缩时间存在差异。一些文件类型，如 MKV 文件（6.656 毫秒）、WAV 文件（5.777 毫秒）、MP4（3.997 毫秒）等音频视频文件，解压缩时间相对较长。这是因为音频视频文件通常具有较大的文件大小和更复杂的解压缩操作。解压缩涉及复杂的解码过程和多个数据通道的处理，因此需要较长的时间。而 TXT 文件（0.067 毫秒）和 XML 文件（0.19 毫秒）显示出较短的解压缩时间，这意味着这些文件相对较容易被解压缩，并且具有较快的解压缩速度。另外，通过比较压缩时间和解压缩时间，看到解压缩时间通常略短于压缩时间。这是因为解压缩操作不需要进行压缩算法的计算过程，只需简单地还原压缩数据，因此通常会更快地完成。
3. 不同文件类型的压缩率存在明显差异。例如，目前人们针对常用音频格式如 MP4 文件（0.00%）、WEBM 文件（0.12%）、MKV 文件（0.71%）等已经进行了不同程度的压缩，但依然包含了大量重复的冗杂信息[7]，

再次进行压缩后的效果不明显。而对于图像文件类型，BMP 图像文件展现出高达 82.40%的压缩率，而 JPG 和 PNG 图像文件的压缩率较低，分别为 0.66%和 0.28%。这是因为 BMP 图像文件本身没有使用压缩算法，而 JPG 和 PNG 图像文件采用了有损和无损压缩算法，所以其压缩率较低。相反，XML 文件(89.51%）和 BIN 文件(86.93%）展现出较高的压缩率，这表明这些文件类型在经过压缩后能够显著减小文件的大小。压缩率的差异反映了不同文件类型的数据特征和压缩算法的适用性。

4. 在结果中也发现了异常情况。对于 MKV、PNG 和 WEBM 这三种文件格式，有些文件压缩后的文件大小反而大于压缩前。这是因为这些文件格式具有一些特性，导致常规的压缩算法难以实现显著的压缩效果。MKV 视频文件格式通常包含已经经过压缩的音频和视频轨道，再次对整个文件进行压缩时，压缩算法难以提供额外的压缩效果，甚至可能使文件大小略微增加。PNG 图像文件格式采用无损压缩算法，旨在保留图像的精确细节和透明度，再次压缩时可能产生一些冗余数据，导致压缩后的文件大小稍大于原始文件。WEBM 多媒体文件格式常用于存储音频和视频数据，并采用了高效的音频编解码器和视频编解码器，再次进行整体压缩时可能无法获得明显的额外压缩效果，甚至压缩后的文件大小略微增加。在网络传输中，对于这些压缩后文件大小反而增加且压缩率极低的文件格式，可以考虑选择不压缩直接传输的方法，以最大程度地减少数据传输的时间和资源消耗。如果带宽充足、网络稳定，并且接收端具备足够的处理能力，那么选择不压缩传输可能也是一种合理的决策。

（2）BMP 和 TXT 格式

在实验中，进一步选取了压缩效果显著且常见的两种文件格式(BMP、TXT)，进行进一步的比较实验。对于每一种文件格式，选择了 7 个不同文件大小的数据集[13][15-16][21-30]进行解压缩操作，并准确记录了解压缩前后文件大小、解压缩时间、压缩率等，以全面评估压缩收益。

本次实验所使用的 BMP 数据集大小约 13.58GB，TXT 数据集大小约 0.23GB。实验数据结果如表 3 所示：

表 3 实验数据结果

参数 类型	数量(个)	压缩前大小(MB)	压缩后大小(MB)	压缩时间(MS)	解压时间(MS)	压缩率(%)
BMP	11686	0.000433	0.000128	0.000154	0.000043	70.44%
BMP	6688	0.01086	0.008109	0.00098	0.000186	58.89%
BMP	15557	0.019508	0.009805	0.001205	0.000188	60.97%
BMP	4049	0.138848	0.102942	0.007094	0.001295	24.48%
BMP	5513	0.148127	0.110608	0.006324	0.001205	25.33%
BMP	15114	0.65723	0.060739	0.011658	0.003121	90.38%
BMP	18971	0.750051	0.132008	0.022156	0.0036	82.40%
TXT	4394	0.000151	0.000084	0.000184	0.000055	44.11%
TXT	2584	0.001119	0.000565	0.000268	0.00006	49.16%
TXT	7760	0.002006	0.001005	0.000302	0.000067	42.28%
TXT	2000	0.003809	0.001796	0.000441	0.000092	50.57%
TXT	19827	0.003972	0.001558	0.000397	0.000078	58.99%

TXT	17561	0.006567	0.002865	0.000473	0.000092	42.27%
TXT	1468	0.01357	0.004375	0.000879	0.000154	67.80%

BMP 格式的文件压缩时间、解压时间如图 2 所示：

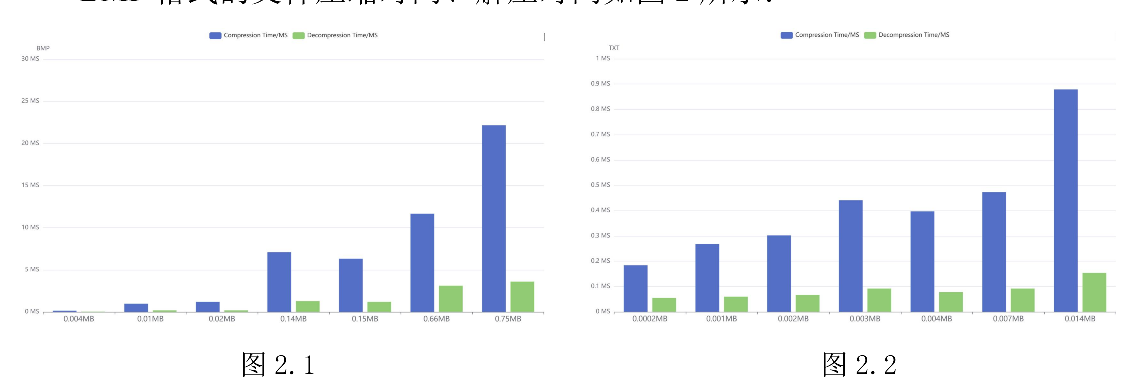


图 2.1　　　　　　　　　　　　图 2.2

图 2.1 显示了 BMP 格式文件的解压时间比较，图 2.2 显示了 TXT 格式文件的解压时间比较。

BMP 格式的文件压缩率如图 3 所示：

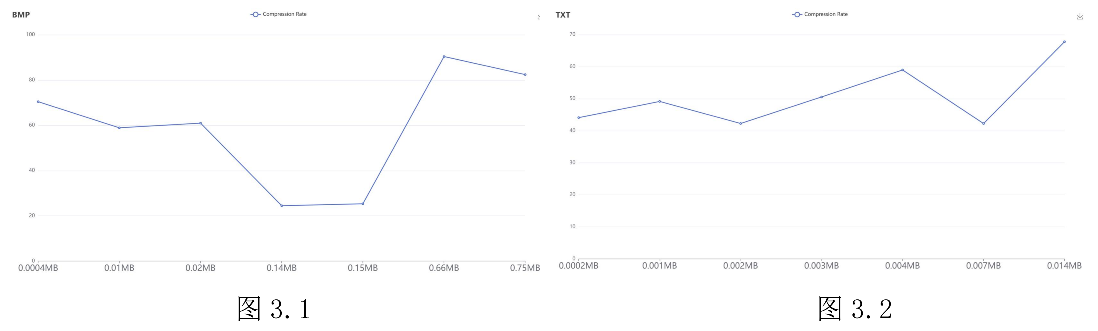


图 3.1　　　　　　　　　　　　图 3.2

图 3.1 显示了 BMP 格式文件的压缩率，图 3.2 显示了 TXT 格式文件的压缩率。

对于 BMP 图像文件格式和 TXT 文本格式，不同大小的文件进行压缩和解压缩操作，压缩率在不同数量级的文件中也存在差异。

（3）HDF5 和 NetCDF 格式

海量数据传输对于高性能计算也具有实践意义，实验选择两种高性能计算文件格式（HDF5 和 NetCDF）进行压缩实验。本次实验所用到的数据集[39-53]大小约共 87GB，其中，HDF5 文件约 51GB，NetCDF 文件约 36GB。实验中准确记录了解压缩前后文件大小、解压缩时间以及压缩率等指标，以便深入了解这些文件格式在数据传输中的性能表现。

HDF5 (Hierarchical Data Format 5)在科学研究、数据分析、高性能计算和可视化等领域得到广泛应用。被用于存储和共享大规模的实验数据、模拟结果、图像数据、遥感数据等。HDF5 格式文件解压缩时间分析如图 4 所示：

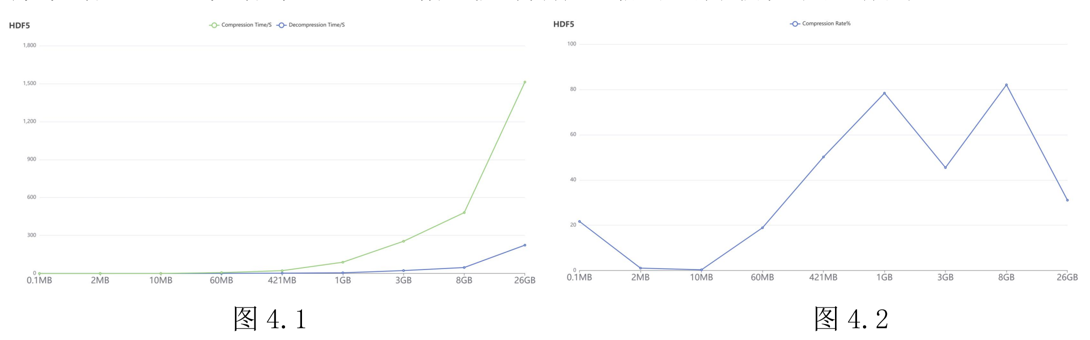


图 4.1　　　　　　　　　　　　图 4.2

图 4.1 显示了 HDF5 格式文件的解压时间和压缩率的比较，图 4.2 显示了 HDF5 格式文件的压缩率。

随着 HDF5 格式文件大小的增加，其解压缩时间呈现逐渐增长的特征。大型文件的解压缩过程涉及到更为繁重的计算任务和资源要求，随着文件规模的扩大，解压缩操作所需的计算资源也随之增加。由于需要处理更大量的数据，解压缩算法在执行阶段必须执行更多的计算操作，这必然导致了解压缩时间的增加。

NetCDF (Network Common Data Form）是一种用于存储、访问和共享科学数据的文件格式。NetCDF 格式文件解压缩时间分析如图 5 所示：

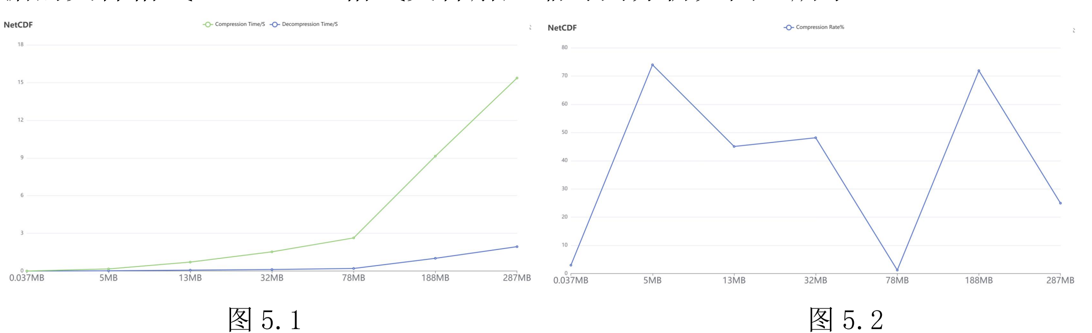


图 5.1　　　　图 5.2

图 5.1 是 NetCDF 格式文件的解压时间和压缩率的比较，图 5.2 是 NetCDF 格式文件的压缩率。

NetCDF 格式文件的解压缩时间通常随着文件大小的增加而增加。大型 NetCDF 文件的解压缩过程涉及更多的磁盘读取操作和计算操作，较大的文件需要更长的时间来从硬盘中读取数据，这会增加解压缩的总体时间。此外，解压缩算法在处理更多数据时可能需要更多的计算操作和内存资源，这也 会导致解压缩时间的增加。

5 结论

5.1 本实验的结论如下：

通过本实验发现，不同类型的文件格式在压缩性能上有明显的差异，有不同的收益。有些文件格式的压缩率较高，压缩效果明显，而有些文件格式的压缩效果较低。对于已经压缩过的文件格式（如 JPG、MP3 等），由于内部使用了特定的压缩算法，可能会导致信息丢失和重复压缩，因此重新压缩效果不佳。另一方面，未经压缩的文件格式，如文本文件和无损图像文件（如 TXT、BMP），往往表现出高压缩率和显著的压缩效果。这是因为这些文件格式有很高的冗余度和可压缩性，通过压缩可以有效地减少文件大小。

5.2 未来展望：

本研究主要是对不同文件格式的压缩性能进行分析，但没有深入研究压缩算法和优化方法的细节。未来的研究可以集中在改进和优化现有的压缩算法，以提高压缩和解压速度，同时保持良好的压缩质量。

参考文献：

[1] 白文超.基于深度学习的海量数据近似计算关键算法的研究与实现[D].哈尔滨工业大学.2022.

[2] 渠开洋.基于改进哈夫曼的上下文数据压缩算法设计与实现[D].北京邮电大学,2017.

[3] 卢冰,刘兴海.利用改进的哈夫曼编码实现文件的压缩与解压[J].科技通报,2013,29(06):22-24.

[4] Mhd.Ali Subada.Comparisonal Analysis Of Even-Rodeh Algorithm Code AndFibonacci Code Algorithm For Text File Compression.Journal Basic Science and Technology.Feb 2022

[5] 李炜. 大数据云存储下海量数据传输完整度控制技术[J]. 吉林大学学报(信息科学版),2019,37(06):682-686.

[6] 郑东旭.卫星导航系统传输带宽优化技术研究[D].沈阳理工大学,2021.

[7] lzma — Compression using the LZMA algorithm — Python 3.11.3 documentation.

[8] Subin An.Kaggle Kerneler.（2021，May）Various Pokemon Image Dataset,Version 1,Retrieved May 25,2023 from Various Pokemon Image Dataset | Kaggle

[9] Theo Viel,Umong Sain,lnnat.(2022,December)RSNA Breast Cancer Detection - 512x512 pngs,Version 1,Retrieved May 26,2023 from RSNA Breast Cancer Detection - 512x512 pngs | Kaggle

[10] DeepMind.(2022,May)Kinetics dataset (5%),Version 1,Retrieved May 26,2023 from Kinetics dataset (5%) | Kaggle

[11] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, Xavier Bresson, EPFL LTS2.(2017,May)FMA: A Dataset For Music Analysis Data Set,Version 1,Retrieved May 26,2023 from UCI Machine Learning Repository: FMA: A Dataset For Music Analysis Data Set.

[12] SyntheaTM.(2021,May)Synthea Dataset Jsons - EHR,Version 1,Retrieved May 26,2023 from Synthea Dataset Jsons - EHR | Kaggle

[13] Dipankar Srirag.Muhammad Navaid.Kaggle Kerneler.(2020,May)Text Classification on Emails, Version 1,Retrieved May 26,2023 from Text Classification on Emails | Kaggle

[14] Linjie Yang, Ping Luo, Chen Change Loy, Xiaoou Tang.A Large-Scale Car Dataset for Fine-Grained Categorization and Verification.Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3973-3981

[15] Md Fantacher lslamA.Matt OP.Guri.(2020,June)Metal Surface Defects Dataset,Version 1,Retrieved June 1,2023 from Metal Surface Defects Dataset | Kaggle.

[16] JinyeongWang.KaggleKerneler.Tannatorn Tantipiriyakul.(2019,June)Alphabet+Numbers.Version 1,Retrieved June 1,2023 from Alphabet+Numbers | Kaggle.

[17] Gzip - GNU Project - Free Software Foundation.

[18] bzip2 : Home (sourceware.org).

[19] https://github.com/madler/zlib

[20] KAIFENG YANG.(2022,June).Heat Sink Surface Defeect Dataset.Version 1,Retrieved June 2,2023 from Heat Sink Surface Defect Dataset | Kaggle.

[21] Gaurav Rajpal.Abdallah Wagih lbrahim.Beyza Nur Nakkas.(2020,June)Leukemia Classification,Version 1,Retrieved June 2,2023 from Leukemia Classification | Kaggle.

[22] Mohammed Aliy.Kaggle Kerneler.(2020,June)Single Cell Conventional Pap Smear Images,Version 1,Retrieved June 2,2023 from Single Cell Conventional Pap Smear Images | Kaggle.

[23] Dustin Ober.(2023,April)Soybean Seeds Classification Dataset,Version 1,Retrieved June 2,2023 from Soybean Seeds| Kaggle.

[24] John Margaronis.Minas Christou.Ergina Kavallieratou.Theodoros Tzouramanis.(2020,June)Handwritten Greek Characters from GCDB,Version 1,Retrieved June 2,2023 from Handwritten Greek Characters from GCDB | Kaggle.

[25] RNA.Kaggle Kerneler.(2020,June)SocioEconomic Data and Applications Center,Version 1,Retrieved June 3,2023 from SocioEconomic Data and Applications Center | Kaggle.

[26] Kaggle Kerneler.Noah Daniels.(2019,June)Sports articles for objectivity analysis,Version 1,Retrieved June 3,2023 from Sports articles for objectivity analysis | Kaggle.

[27] kobeshigaidaicorpus.Sami Hirata.Kaggle Kerneler.(2020,June)NICT Japanese Learners of English 4.1,Version 1,Retrieved June 3,2023 from NICT Japanese Learners of English 4.1 | Kaggle.

[28] Hsankesara.Paul Mooney.stpete_ishii.(2019,June)CVPR 2019 Papers,Version 1,Retrieved June 3,2023 from CVPR 2019 Papers | Kaggle.

[29] MihxSP.Kaggle Kerneler.Mike Klemin(2020,June)Kinopoisk's movies reviews,Version 1,Retrieved June 3,2023 from Kinopoisk's movies reviews | Kaggle.

[30] Alien.Huan YEH.Kaggle Kerneler.(2021,June)vinbigdata txt yolov5Version 1,Retrieved June 3,2023 from vinbigdata txt yolov5 | Kaggle.

[31] 刘晓燕,许志伟,李文越等.面向高效边缘计算的可分类数据压缩传输机制[J].软件导刊,2022,21(11):38-43.

[32] 李明,尹时松,张宁等.基于稀疏表示的阵列声波测井仪数据无损压缩传输方法[J].测控技术,2022,41(05):106-112.

[33] 王巨龙,张怀柱,玄金鹏.宽频带地震仪数据采集及压缩传输的研究[U].现代电子技术,2020,43(04):100-103.

[34] 闫亮,李永斌.计算机网络传输中有效压缩数据的方法研究[J].通讯世界,2016,(15):20-21.

[35] 杨敬锋,张南峰,李勇等.基于改进 Huffman 编码的农机作业数据传输压缩方法[J].农业工程学报,2014,30(13):153-159.

[36] 彭冲. 面向传感器网络大数据传输应用的数据压缩与传输优化算法的研究与应用[D].电子科技大学,2014.

[37] 马兴明,董成,毛新宇等.基于状态估计的海量多元异构智能电网数据压缩存储方法[J].电机与控制应用,2023,50(02):67-72+81.

[38] 王鹤,李石强,于华楠等.基于分布式压缩感知和边缘计算的配电网电能质量数据压缩存储方法[J].电工技术学报,2020,35(21):4553-4564.

[39] SUNGHO SHIM.（2022，Augest）Tiny ImageNet (HDF5),Version 1,Retrieved June 7,2023 from Tiny ImageNet (HDF5) | Kaggle.

[40] Kaggle Kerneler. Benedict Wilkins. Limon Halder. (2020, June). MNIST - HDF5, Version 1, Retrieved June 7, 2023 from MNIST - HDF5 | Kaggle.

[41] MUHAMMAD IRFAN AZAM. (2022, June). AMEX HDF5 - Last Statement - Train Only, Version 1, Retrieved June 7, 2023 from AMEX HDF5 - Last Statement - Train Only | Kaggle.

[42] Kagle Kerneler. Olga Belitskaya. (2020, June). Quick, Draw! Model Weights for Doodle Recognition, Version 1, Retrieved June 8, 2023 from Quick, Draw! Model Weights for Doodle Recognition | Kaggle.

[43] Valentyn Sichkar. (2021, June). Traffic Signs 1 million images for Classification, Version 1, Retrieved June 8, 2023 from Traffic Signs 1 million images for Classification | Kaggle.

[44] K Scott Mader. 孙健行 James_97_soton. Wouter van Amsterdam. (2017, June). Lung Nodule Malignancy, Version 1, Retrieved June 8, 2023 from Lung Nodule Malignancy | Kaggle.

[45] ladyofateele. Habineza. Elena Cuoco. (2019, June). The Gravitational Waves Discovery Data, Version 1, Retrieved June 8, 2023 from The Gravitational Waves Discovery Data | Kaggle.

[46] Marco Polo. Richard Kuo. Asmaa. (2020, June). Brain Tumor Segmentation(BraTS2020), Version 1, Retrieved June 8, 2023 from Brain Tumor Segmentation(BraTS2020) | Kaggle.

[47] MC1138.（2022, October）. GISTEMP seasonal trends, Version 1, Retrieved June 8, 2023 from GISTEMP seasonal trends | Kaggle.

[48] Baris Dincer. (2021, June). CLIMATE CHANGE MADAGASCAR-TURKEY /NASA, Version 1, Retrieved June 8, 2023 from CLIMATE CHANGE MADAGASCAR - TURKEY / NASA | Kaggle.

[49] Gabriel Preda. Baris Dincer. Kaggle Kerneler. (2020, June). EarthData MERRA2 Co, Version 1, Retrieved June 8, 2023 from EarthData MERRA2 CO | Kaggle.

[50] Paula Romero Jure. Kaggle Kerneler(2020, June). GOES_L1, Version 1, Retrieved June 8, 2023 from GOES_L1 | Kaggle.

[51] Abhinav Sharma. (2021, June). ERA data files - daily(122) for selected domain, Version 1, Retrieved June 8, 2023 from ERA data files - daily(122) for selected domain | Kaggle.

[52] Vincentive Larmet. Kaggle Kerneler. (2020, June). ERA interim wind data in Puerto Rico, Version 1, Retrieved June 8, 2023 from ERA interim wind data in Puerto Rico | Kaggle.

[53] Parichat Wetchayont. (2021, January). OSTIA SST Asia, Version 1, Retrieved June 8, 2023 from OSTIA SST Asia | Kaggle.

(通讯作者：秦广军

E-mail:zhtguangjun@buu.edu.cn)

作者贡献声明

杨含：提出研究思路，设计研究方案，进行实验，论文起草。

胡永庆：采集、清洗和分析数据。

秦广军：论文最终版本修订。